<u>REMARKS</u>

Claims 1-45 are currently pending in the subject application and are presently under consideration. Claims 1, 16, 32, 33, and 41-45 have been amended, as shown on pages 2-13 of the Reply.

Favorable reconsideration of the subject patent application is respectfully requested in view of the comments and amendments herein.

I.   **Rejection of Claims 16, 23, 30, 32-35, 37, 38, 41, 42, 43, and 45 Under 35 U.S.C. §103(a)**

Claims 16, 23, 30, 32-35, 37, 38, 41, 42, 43, and 45 stand rejected under 35 U.S.C. §103(a) as being unpatentable over Oppenheim (US 5,734,905) in view of the Mac OS 8.5 operating system, as described by "The Macintosh Bible, 7th Edition" written by Sharon Zardetto Aker (Macintosh reference). Withdrawal of this rejection and allowance of the subject claims is respectfully requested for at least the following reason. Oppenheim and the Macintosh reference, alone or in combination, fail to disclose, teach, or suggest each and ever element of applicants' claimed invention.

> To reject claims in an application under §103, an examiner must establish a *prima facie* case of obviousness. A *prima facie* case of obviousness is established by a showing of three basic criteria. First, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or to combine reference teachings. Second, there must be a reasonable expectation of success. Finally, the prior art reference (or references when combined) must teach or suggest all the claim limitations. *See* MPEP §706.02(j). The teaching or suggestion to make the claimed combination and the reasonable expectation of success ***must both be found in the prior art*** and not based on applicant's disclosure. *See In re Vaeck,* 947 F.2d 488, 20 USPQ2d 1438 (Fed. Cir. 1991).

The subject invention generally relates to mapping a source object with a target object to enable documents or items associated with a first schema or structure to be translated to accord to a second schema or structure. More particularly, the claimed invention relates to function objects, which may be used for defining a mapping between

a source object and a target object. Function objects allow unsophisticated users to graphically define and compile a mapping between objects having disparate schema without having to create functional computer-executable scripts, thus providing significant advantages over conventional methods (*e.g.*, an ordinary user need not have extensive knowledge in computer programming to generate a mapping between two objects). To that end, independent claim 16 recites ***a source object node displayed hierarchically amongst a plurality of source object nodes conforming to a first schema and a target object node displayed hierarchically amongst a plurality of target object nodes conforming to a second schema***. Independent claims 32, 33, 41, 42, 43, and 45 recite similar elements. The combination of Oppenheim and the Macintosh reference fails to disclose or suggest such claimed aspects

       Oppenheim discloses systems and/or methods that enable a user to link applications in a manner that causes data flowing from one application into a second application linked thereto to be processed, and then route the processed data to a third application for further processing (*See* Oppenheim, col. 8, lines 34-36). In an example of the aforementioned linking, Oppenheim discloses that an application, an analog-to-digital converter object (A/D converter), can be linked with another application, a signal processor object, thereby causing the output of the A/D converter object to be employed as input to the signal processor object. (*See* Oppenheim, col. 8, lines 38-53) Oppenheim further discloses that a third application, a filter object, can be linked to an output port of the signal processor object, so that output of the signal processor object is utilized as input to the filter object (*See* Oppenheim, col. 8, lines 54-57).

       The applications, however, are ***not*** displayed hierarchically, as such applications can be operated independent of one another (*See* Oppenheim, fig. 8). Input and output ports that may be associated with an application described in Oppenheim are additionally not ***displayed hierarchically*** as claimed. Rather, the input ports and output ports (which applicants' representative submits are not nodes of a source object) have no hierarchical relationship therebetween, much less a ***source object node that conforms to a first*** ***<u>document structure</u> and a target object node that conforms to a second <u>document</u>*** ***<u>structure</u>*** as recited in claim 43, for example. Rather, Oppenheim teaches that various applications can be linked together, and makes no mention of different schemas, much

less different document structures, much less source nodes or target notes that conform to different schemas as is recited in these claims. For example, the first-linked application (*e.g.,* A/D converter) does not include a plurality of nodes conforming to a first schema, and a last-linked application (*e.g.,* filter object) does not include a plurality of object nodes conforming to a second schema as claimed. Oppenheim does not contemplate such schemas as the teachings therein are directed towards manipulating data as it passes through each application in real-time, for instance, to modify digital music.

The Macintosh reference fails to make up for the aforementioned deficiencies of Oppenheim. The Macintosh reference is cited to illustrate that file folders can be arranged in a hierarchical manner. The Macintosh reference, like Oppenheim, however fails to disclose, teach, or suggest *a source object node displayed hierarchically amongst a plurality of source object nodes conforming to a first schema and a target object node displayed hierarchically amongst a plurality of target object nodes conforming to a second schema*.

Further, there would be no suggestion or motivation for one of ordinary skill in the art to combine or modify the teachings of the Oppenheim reference with the teachings of the Macintosh reference. The Oppenheim reference relates to dragging one graphical object over a second graphical object (*e.g.,* "slapping") in order to perform a transformation of the second graphical object and linking different applications together such that data flows through the applications in real-time. Again, the Macintosh reference relates to organizing file folders. There is nothing in the references to suggest that there are "nodes", let alone that it would be beneficial to have a hierarchical arrangement of nodes to facilitate "slapping" one graphical object onto another graphical object. Thus, even assuming the teachings of Oppenheim were not otherwise deficient as to certain elements of the claimed invention the teachings of the prior art, taken as a whole, do not make obvious the claimed invention.

In addition, independent claims 33 and 41, relate to *creating a function object*. Claim 33, and similarly claim 41, recite another limitation: *creating a script component having computer-executable instructions for performing a function using the user interface; creating a graphical component associated with the function having at least one input and at least one output*.

Oppenheim is silent as to a *function object* as claimed, and thus Oppenheim cannot disclose, teach, or suggest a method of making a function object. Oppenheim discusses modifying an application object by "slapping" another application object on top of the first application object, and making this modified application object its own distinct object. (*See* Oppenheim, col. 7, lines 21-28) Nowhere does Oppenheim state that this results in the creation of a script component related to this modified application object, or that such object would have inputs or outputs.

The Examiner also asserts that Oppenheim discloses certain application programs that allegedly can be used to create an object (*See* Oppenheim, col. 7, lines 1-5). However, Oppenheim does not disclose whether such programs can create a function object comprising creating a script component having computer-executable instructions for performing a function using the user interface, and creating a graphical component associated with the function having at least one input and at least one output.

Further, regarding claims 33 and 41, as well as independent claims 43 and 45, such claims recite *an interface component that provides the script component to a compiler in the mapping tool*. Oppenheim is silent as to an interface component that provides the script component to a compiler in the mapping tool. Rather, any scripts that can operate in connection with transforming an application object are pre-existent, and their creation is not disclosed. (*See* Oppenheim, col. 5, lines 46-51) Oppenheim does not disclose a compiler, let alone *providing a script component to a compiler in a mapping tool*. Moreover, Oppenheim does not disclose, teach, or suggest creating a mapping, and therefore fails to disclose any sort of *mapping tool*.

In view of at least the foregoing, it is readily apparent that Oppenheim and the Macintosh reference do not disclose, teach, or suggest all of the elements of the claimed invention, as recited in independent claims 16, 32, 33, 41, 42, 43 and 45 (and all associated claims that depend therefrom). Accordingly, it is believed that the subject claims are in condition for allowance and that this rejection should be withdrawn.

II.   Rejection of Claims 1-15, 17-22, 24-27, and 44 Under 35 U.S.C. §103(a)

Claims 1-15, 17-22, 24-27, and 44 stand rejected under 35 U.S.C. §103(a) as being unpatentable over Oppenheim in view of the Macintosh reference, and further in

view of Microsoft's "Component Object Model Specification" (Microsoft's COM Specification). This rejection should be withdrawn for at least the following reason. Oppenheim, the Macintosh reference, and Microsoft's COM Specification, alone or in combination, fail to disclose, teach, or suggest each and every element recited in the subject claims.

Independent claim 1 (and similarly independent claim 44) recites *a source object node being one of a plurality of hierarchically displayed nodes conforming to a first schema and a target object node being one of a plurality of hierarchically displayed nodes conforming to a second schema, disparate from the first schema*. For the reasons stated, *infra*, Oppenheim and the Macintosh are silent with respect to such claimed aspects. Further, Microsoft's COM Specification fails to cure the aforementioned deficiencies of Oppenheim and the Macintosh reference with respect to claim 1 (and similarly claims 16 and 44). Accordingly, this rejection should be withdrawn.

III.   Rejection of Claims 28, 29 and 31 Under 35 U.S.C. §103(a)

Claims 28, 29 and 31 stand rejected under 35 U.S.C. §103(a) as being unpatentable over Oppenheim in view of in view of the Macintosh reference, and further in view of Jordan (US 5,778,227). This rejection should be withdrawn for at least the following reason. Oppenheim, the Macintosh reference, and Jordan, alone or in combination, fail to disclose, teach, or suggest all the limitations of the subject claims. The subject claims depend upon independent claim 16. Jordan fails to cure the aforementioned deficiencies of Oppenheim and the Macintosh reference with respect to claim 16. Therefore, withdrawal of this rejection is respectfully requested.

IV.   Rejection of Claims 36, 39 and 40 Under 35 U.S.C. §103(a)

Claims 36, 39 and 40 stand rejected under 35 U.S.C. §103(a) as being unpatentable over Oppenheim in view of the Macintosh reference, and further in view of Faustini (US 6,496,870). This rejection should be withdrawn for at least the following reason. Faustini fails to make up for the deficiencies of Oppenheim and the Macintosh reference with respect to claim 33 (upon which the subject claims depend). Therefore, this rejection should be withdrawn.

CONCLUSION

The present application is believed to be in condition for allowance in view of the above comments and amendments. A prompt action to such end is earnestly solicited.

In the event any fees are due in connection with this document, the Commissioner is authorized to charge those fees to Deposit Account No. 50-1063[MSFTP128US].

Should the Examiner believe a telephone interview would be helpful to expedite favorable prosecution, the Examiner is invited to contact applicants' undersigned representative at the telephone number below.

Respectfully submitted,

AMIN, TUROCY & CALVIN, LLP

/Himanshu S. Amin/
Himanshu S. Amin
Reg. No. 40,894

AMIN, TUROCY & CALVIN, LLP
24TH Floor, National City Center
1900 E. 9TH Street
Cleveland, Ohio 44114
Telephone (216) 696-8730
Facsimile (216) 696-8731